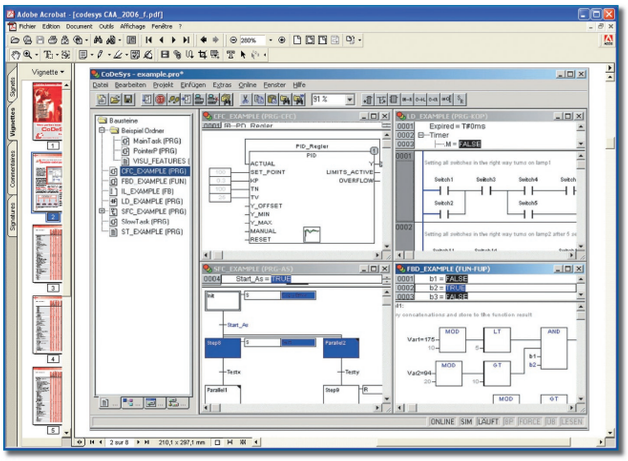
Plus loin que la CEI

Après une bonne dizaine d'années de présence dans le monde des automatismes, la société 3S-Smart Software a su s'imposer. Le nom de cette entreprise implantée en Allemagne vous est peut-être inconnu, pourtant le nom du produit vous est familier : Codesys. Voici la version 3.0, avec pas mal d'évolutions dont l'implantation de la technologie .Net de Microsoft, la programmation orientée objet ou la possibilité de programmer un réseau de contrôleurs en un seul et unique projet.

En quelques années, le produit est en phase de passer du statut d'interface de programmation pour composants d'automatismes à celui d'outil logiciel pour l'automaticien avec l'apport d'une supervision et d'un contrôle d'axes intégrés. D'un développement pour les fabricants d'automates programmables allemands, la société est devenue leader, avec notamment des ouvertures vers l'industrie de l'embarqué, de la GTB/GTC, des engins mobiles, des fabricants de machines... de quoi atteindre les 4,7 millions d'euros de chiffre d'affaires en 2006 avec 45 personnes, la France représentant, en chiffre d'affaires, le deuxième marché le plus important, après l'Allemagne.

Cette suite logicielle est composée de trois grandes parties, l'interface utilisateur, la communication et la cible.

L'interface de développement comprend l'outil de programmation avec toutes les fonctions en et hors ligne, les compilateurs, ainsi que les composants additionnels pour la configuration, supervision etc. Vont venir se rajouter des composants optionnels comme la commande de mouvements, mais aussi des extensions spécifiques client (Automation Platform) qui ouvrent le système vers les appareils sur bus de terrain non programmables (comme la configuration de capteurs ou de vannes).



La communication entre le système de développement et la cible hardware est basée sur le serveur Codesys Gateway, sur lequel est installé un serveur OPC. Enfin, pour pouvoir utiliser un appareil avec le logiciel, un runtime doit y être installé.

Le noyau de la suite logicielle reste le système de programmation de contrôleurs, tous les éditeurs définis dans la norme sont à disposition pour la programmation : texte structuré (ST), diagramme fonctionnel en séquences (SFC), boîtes fonctionnelles (FBD), langage ladder (LD), liste d'instructions (IL) et boîtes fonctionnelles graphiques libres (CFC).

Au fil des ans, le logiciel a engrangé certaines demandes des utilisateurs, c'est ainsi que vous pouvez changer l'affichage d'un programme développé en liste d'instructions (IL), ladder (LD) ou boîtes fonctionnelles (FBD) dans n'importe lequel de ces trois langages, en mode hors ligne comme en mode en ligne. Les éléments de syntaxe peuvent être entrés directement à l'aide du clavier ou bien tirés à partir d'une fenêtre outil dans l'éditeur. De plus, une aide intelligente à la saisie et une fonction, du nom d'IntelliSense, aide à la programmation. Les éditeurs textuels permettent d'ouvrir et de fermer des ensembles syntaxiques cohérents (ex. : conditions-IF, boucles-FOR, classes de variables, etc.).

Configuration de projets

Il est possible de relier des contrôleurs de tous genres et de différents fabricants. Chaque contrôleur peut être programmé séparément ou configuré dans le réseau de contrôleurs via ses interfaces de communication. Le projet entier, comprenant différents programmes applicatifs pour des cibles variées, peut être chargé et exécuté à l'aide d'une seule commande.

Chaque contrôleur contient une ou plusieurs applications pour que, si besoin est, plusieurs utilisateurs puissent travailler simultanément sur ce contrôleur. La configuration des tâches permet de partager l'application en différentes couches temporelles synchronisées avec les E/S. Les composants d'E/S sont également configurés dans l'arbre de projet. Les modules fonctionnels créés dans le projet peuvent être attribués à un appareil spécifique ou mis à disposition de l'ensemble des appareils du projet. Il est également possible

de déposer des modules prédéfinis dans des bibliothèques et d'intégrer ces dernières dans un projet.

Les compilateurs intégrés génèrent du code machine à partir des applications CEI 61131-3 et les familles de processeurs 8, 16 et 32 bits les plus utilisés dans l'industrie sont supportées. Les compilateurs entrent en action dès la saisie du code : les erreurs de syntaxe, messages d'alerte et informations sur l'application sont donnés avant la compilation, de quoi réagir avant de télécharger l'application.

En pratique, les compilateurs supportent les données digitales (Booléennes) et analogues comme entier, bit variable, date et heure (TIME, DATE, etc.), virgules flottantes, chaînes et UNIONs. En plus des capacités des langages CEI, les compilateurs proposés ont des extensions permettant une programmation proche d'une programmation système, comme accès aux bits, accès des circuits prétraités (arrays) aux chaînes de caractères (strings) et pointeurs

Programmation orientée objet

Avec la version 3.0, c'est l'ouverture vers le monde Microsoft et .Net, avec la possibilité de choisir de programmer un contrôleur orienté objet, avec la structure des langages comme Java ou C++ : classes, interfaces, méthodes, hérédité, polymorphisme. Les modules fonctionnels CEI ont été étendus pour les rendre opérationnels avec d'autres composants d'ingénierie comme la supervision ou la configuration des bus de terrain.

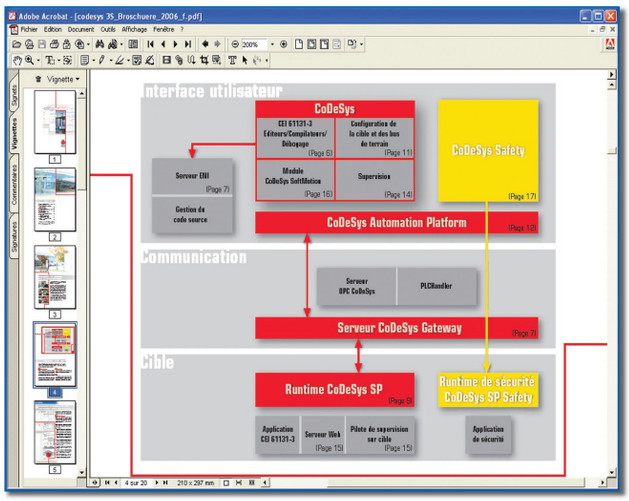
Cette programmation orientée objet va permettre, par exemple, la réutilisation d'une partie d'une application ou la programmation simultanée par plusieurs développeurs.

Pour débogger, l'utilisateur télécharge le code machine, généré à partir de l'application, vers la cible matérielle d'un clic de souris. Les valeurs des variables déclarées sont affichées dans le code, et peuvent être modifiées ou forcées.

Des points d'arrêt et un pilotage d'exécution ligne par ligne, sont utilisés pour retrouver les erreurs. Ces points d'arrêt sont configurables pour pouvoir cibler la recherche d'erreurs. En mode cycle unique, il est possible de suivre l'exécution de chaque modification durant un cycle complet. Lors de modifications dans une application, seules ces modifications seront compilées, téléchargées et activées sans l'arrêt du contrôleur ni la perte des valeurs des variables. C'est également le cas pour les modifications sur plusieurs modules, sur les variables ou les types de données. Cette fonctionnalité se nomme changement en ligne.

Pour pouvoir programmer un appareil avec Codesys, il est nécessaire de configurer l'interface logicielle correspondante avec le runtime. Cette implémentation transforme un appareil embarqué ou basé sur PC, en un contrôleur selon la CEI 61131-3 programmable.

Le runtime est adapté à la plateforme hardware spécifique grâce à une configuration et à des modules customisés, comme les interfaces E/S, le système d'exploitation et les caractéristiques spécifiques du matériel comme watchdog, interrupteur marche/arrêt, etc.



Evolutions récentes et futures

Comme nous l'avons évoqué, le logiciel a suivi les évolutions du monde des automatismes, notamment avec plusieurs interfaces de communication, que ce soit PLC Handler ou Serveur OPC.

PLCHandler est une bibliothèque logicielle, indépendante de la plate-forme matérielle, pour la communication entre différents logiciels clients (par ex. outils de supervision) et les contrôleurs Codesys. Quant au Serveur OPC, il est disponible comme serveur autonome, indépendamment de la plate-forme CPU, de l'OS ou du protocole de communication du contrôleur.

Communication

Pour chapeauter le tout, 3S-Smart Software Solutions propose son Automation Platform, une plate-forme de développement basée sur .Net et permettant de faire évoluer le système de programmation. C'est également une structure complète pour l'intégration d'environnements de développement pour des réseaux de contrôleurs industriels.

Un outil qui permettra à l'automaticien de faire évoluer CoDeSys 3.0 avec ses propres composants – comme par exemple l'intégration d'un nouveau langage de programmation ou d'un nouveau bus de terrain.

Autre utilisation possible de ce logiciel, c'est son emploi comme outil de supervision, sachant que la connexion vers l'extérieur reste possible avec OPC. Aucun outil additionnel n'est nécessaire pour superviser les données d'un contrôleur programmable, un éditeur de masques de visualisation étant disponible dans le système de programmation.

Dernière extension logicielle possible, le contrôle de sécurité intégré, qui facilite le développement et la certification d'automates de sécurité selon la norme SIL3, l'acceptation par le TUV et la BGIA est prévue pour le milieu de cette année.

Pour coller aux solutions Ethernet temps réel, le protocole Ethercat vient d'être implémenté. Le maître Ethercat, sous forme d'une bibliothèque, peut être porté vers n'importe quel appareil programmable possédant une interface Ethernet, l'exécution de la pile maîtresse peut être synchronisée avec les tâches CEI, la fonction mapping Ethercat étant réalisée par l'image des entrées-sorties. ❑